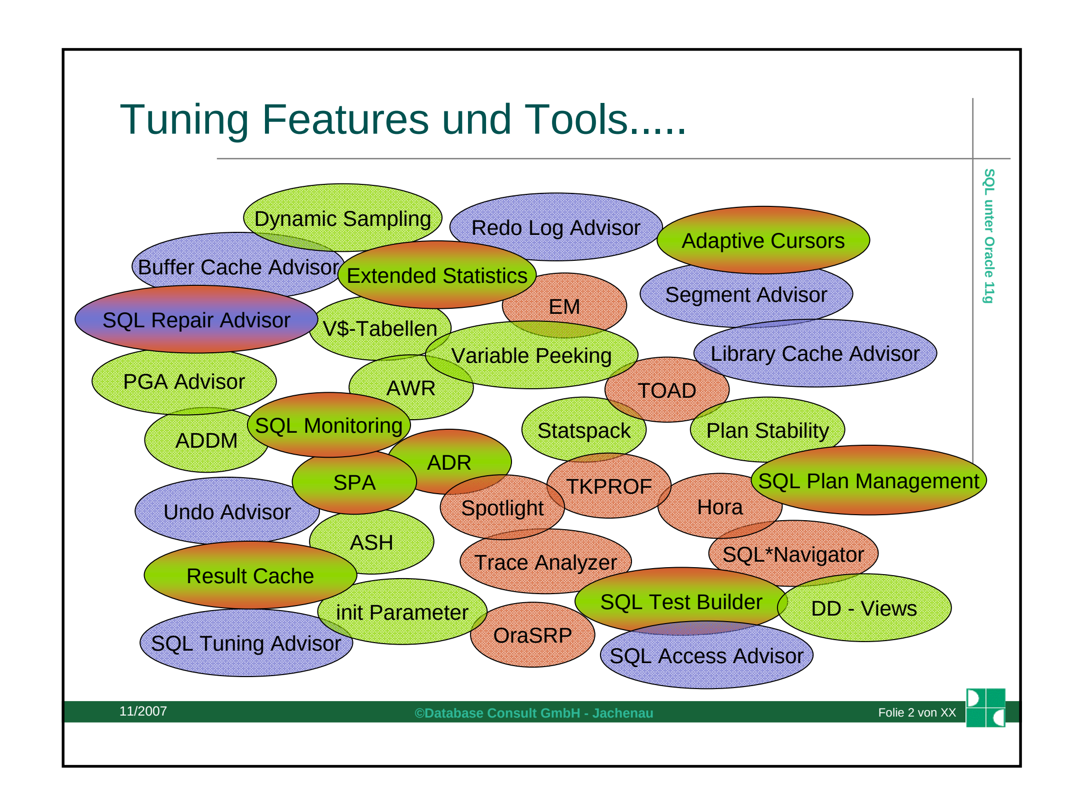
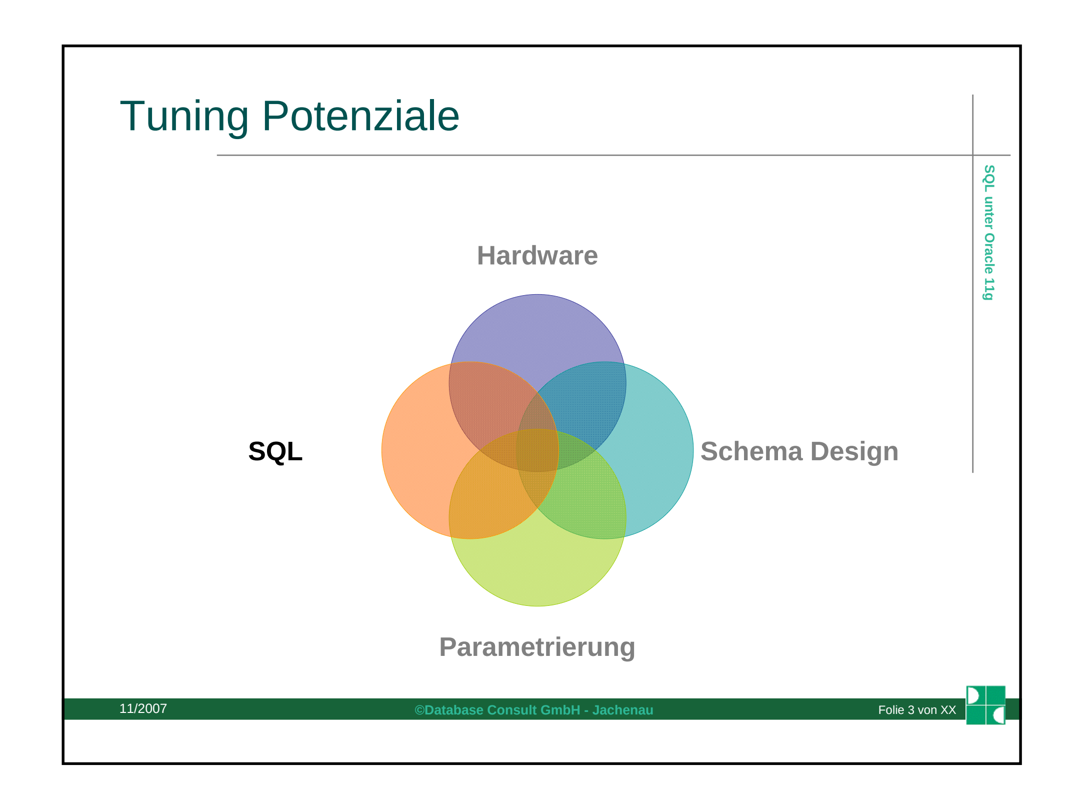
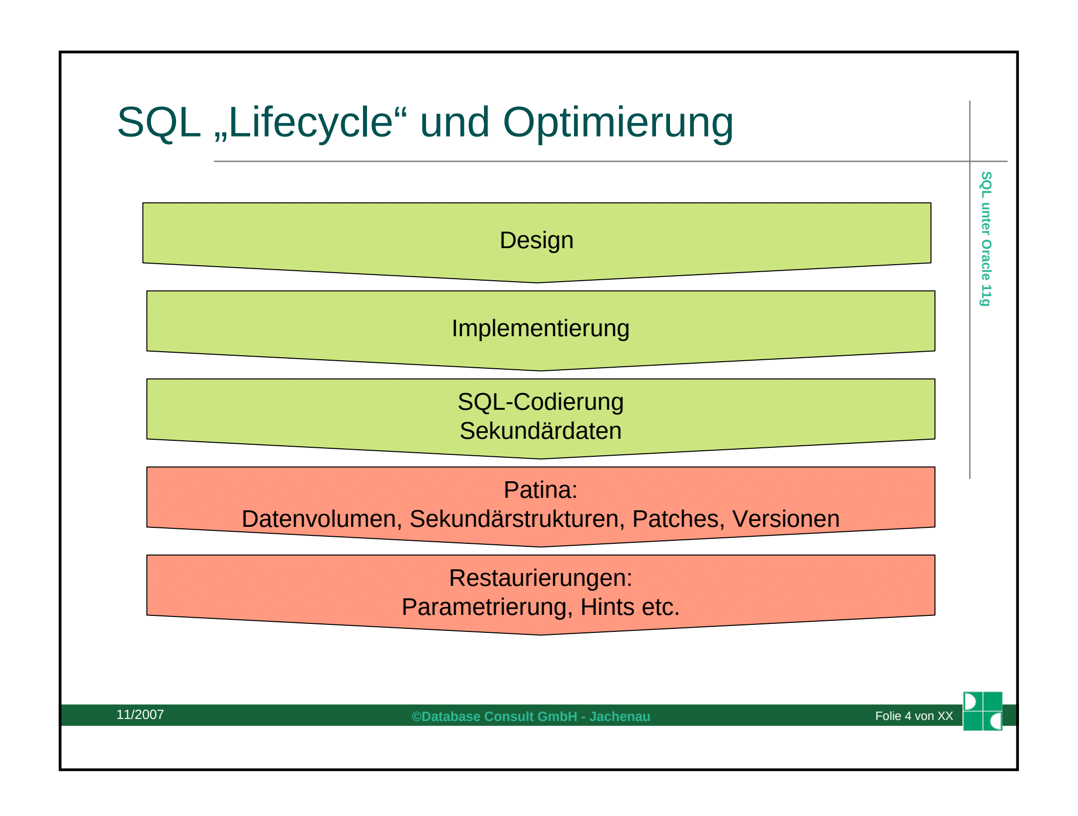
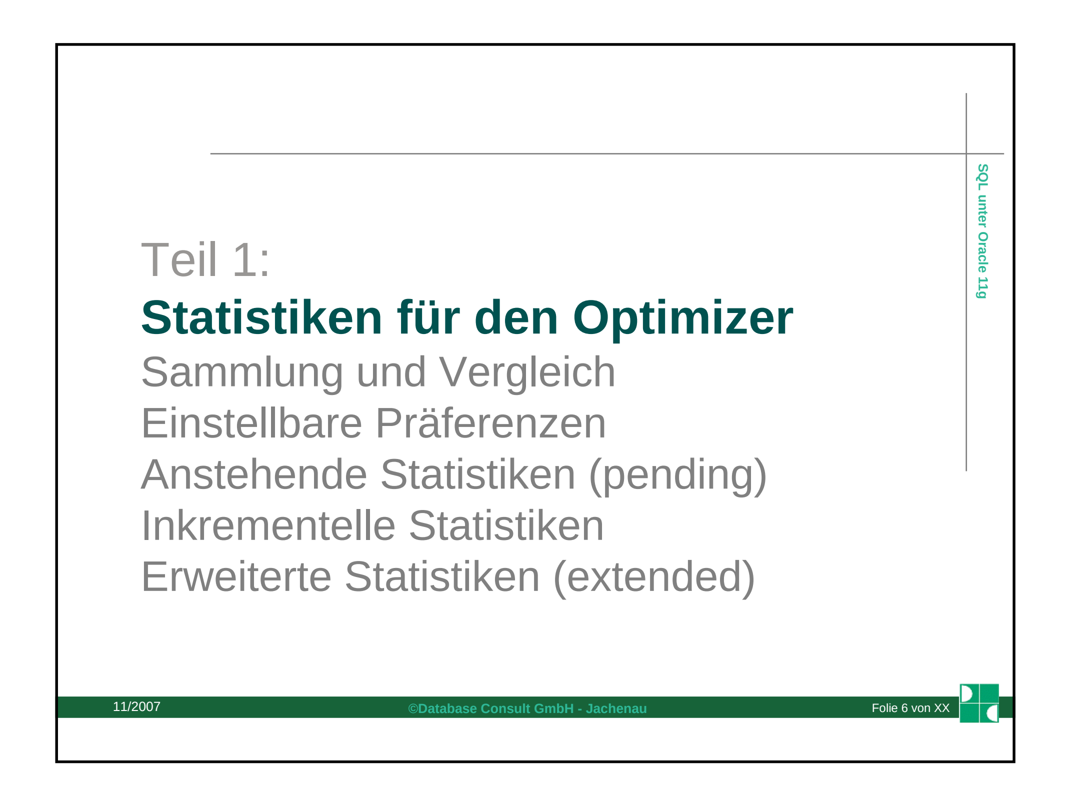

CONSULT
DATABASE
SQL unter Oracle 11g
Monitoring, Tuning, Troubleshooting
Dr. Günter Unbescheid
Database Consult GmbH - Jachenau

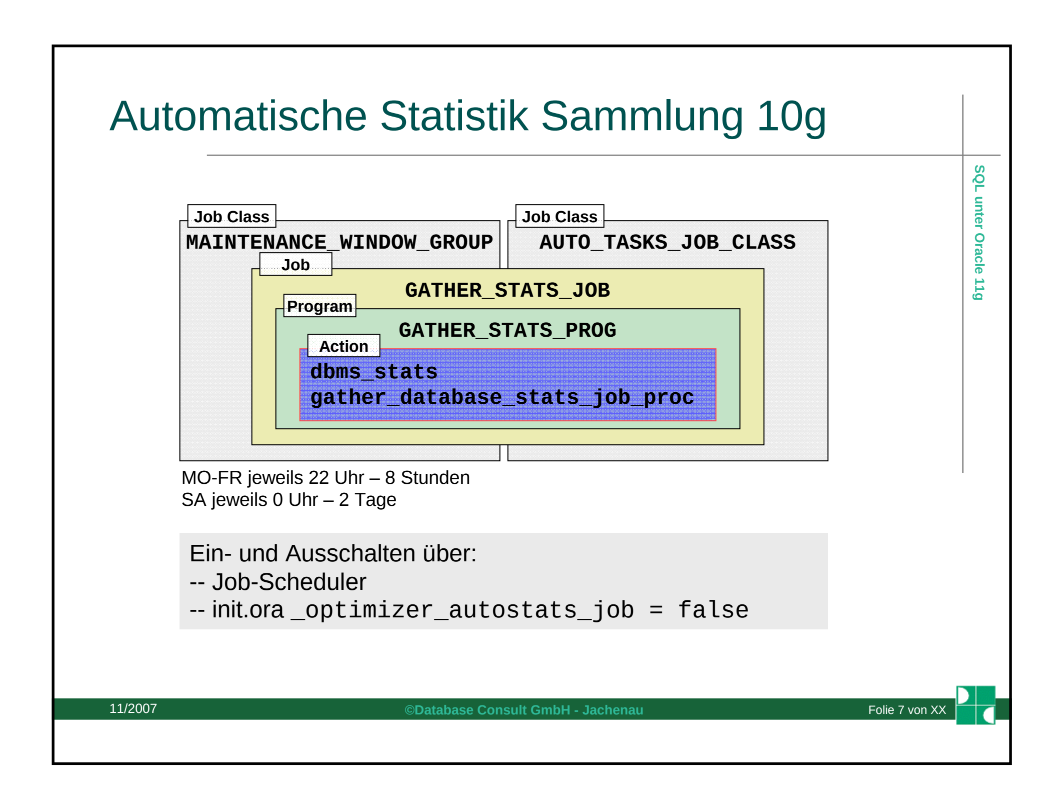

Tuning Features und Tools.....
Dynamic Sampling
Redo Log Advisor
Adaptive Cursors
Buffer Cache Advisor
Extended Statistics
EM
Segment Advisor
SQL Repair Advisor
V$-Tabellen
Variable Peeking
Library Cache Advisor
PGA Advisor
AWR
TOAD
ADDM
SQL Monitoring
Statspack
Plan Stability
ADR
SPA
TKPROF
SQL Plan Management
Undo Advisor
Spotlight
Hora
ASH
Trace Analyzer
SQL*Navigator
Result Cache
SQL Test Builder
DD - Views
init Parameter
OraSRP
SQL Tuning Advisor
SQL Access Advisor
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 2 von XX

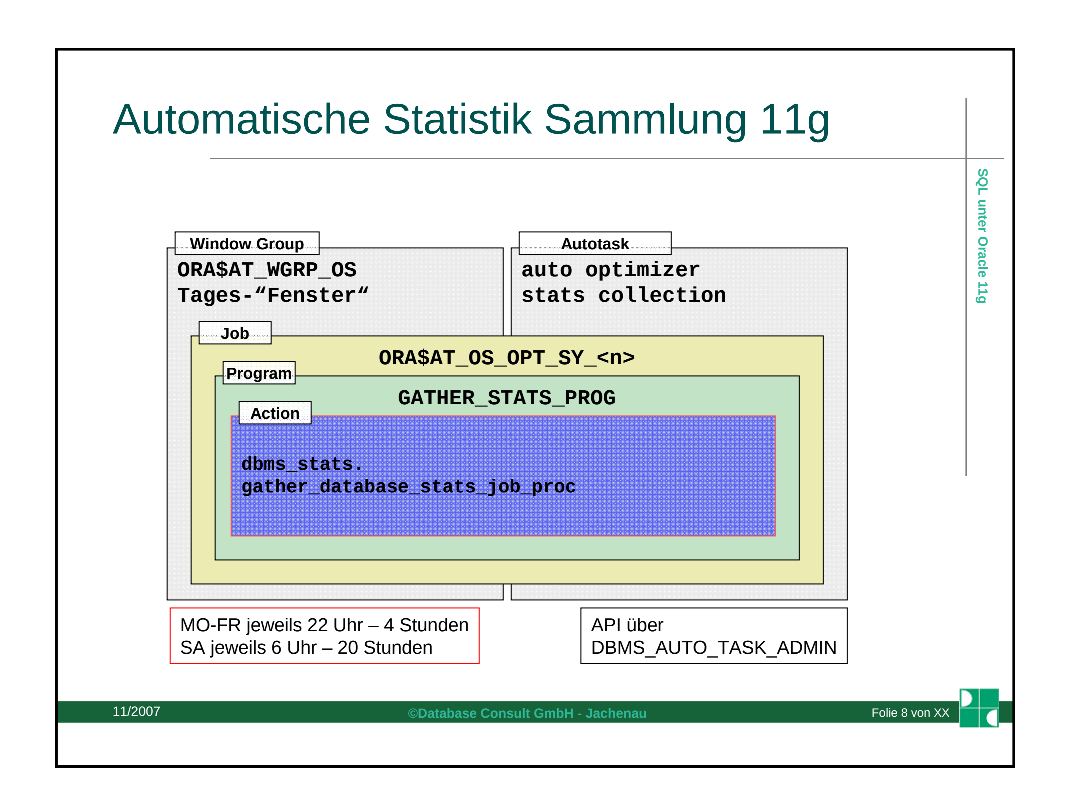

Tuning Potenziale
SQL unter Oracle 11g
Hardware
SQL
Schema Design
Parametrierung
11/2007
©Database Consult GmbH - Jachenau
Folie 3 von XX


SQL „Lifecycle“ und Optimierung
SQL unter Oracle 11g
Design
Implementierung
SQL-Codierung
Sekundärdaten
Patina:
Datenvolumen, Sekundärstrukturen, Patches, Versionen
Restaurierungen:
Parametrierung, Hints etc.
11/2007
©Database Consult GmbH - Jachenau
Folie 4 von XX


Agenda
• Statistiken für den Optimizer
• Adaptive Cursor
• SQL Plan Management
• Real Time SQL Monitoring
• Result Cache
• Automatic SQL Tuning
• SQL Performance Analyzer
• SQL Repair Advisor
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 5 von XX


Teil 1:
Statistiken für den Optimizer
Sammlung und Vergleich
Einstellbare Präferenzen
Anstehende Statistiken (pending)
Inkrementelle Statistiken
Erweiterte Statistiken (extended)
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 6 von XX


Automatische Statistik Sammlung 10g
Job Class
MAINTENANCE_WINDOW_GROUP
Job Class
AUTO_TASKS_JOB_CLASS
Job
GATHER_STATS_JOB
Program
GATHER_STATS_PROG
Action
dbms_stats
gather_database_stats_job_proc
MO-FR jeweils 22 Uhr – 8 Stunden
SA jeweils 0 Uhr – 2 Tage
Ein- und Ausschalten über:
-- Job-Scheduler
-- init.ora _optimizer_autostats_job = false
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 7 von XX

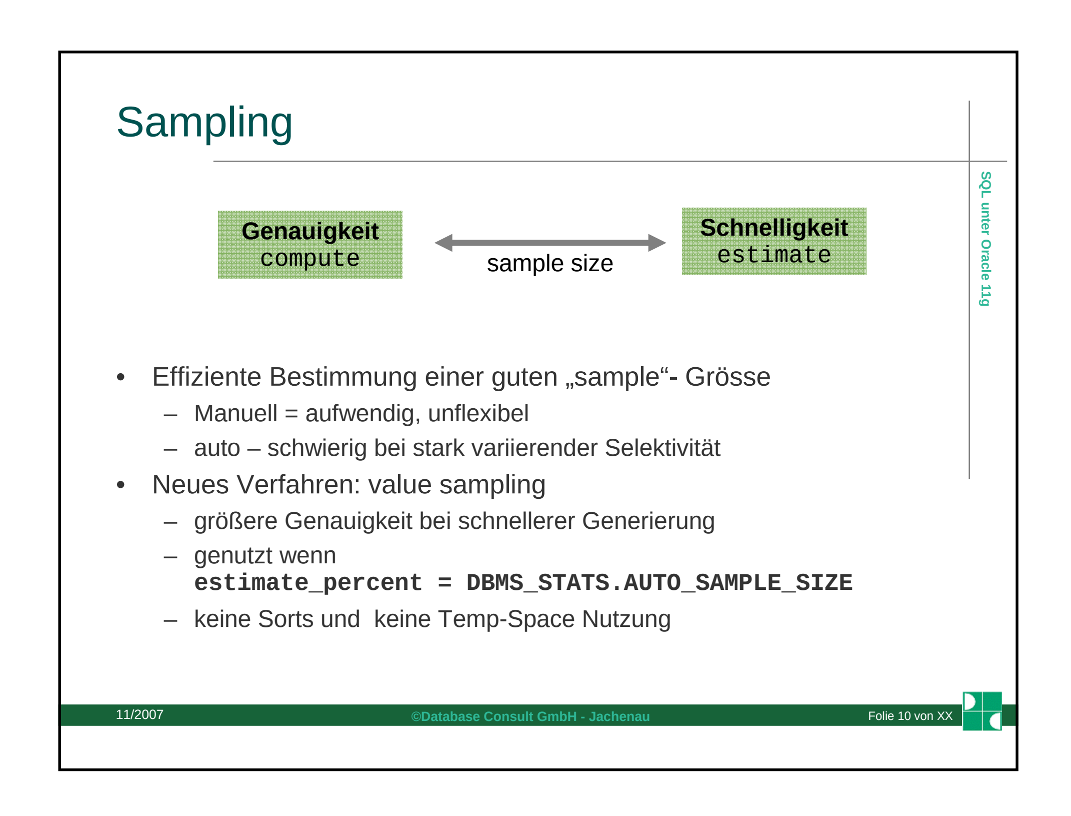

Automatische Statistik Sammlung 11g
Window Group
ORA$AT_WGRP_OS
Tages-“Fenster“
Autotask
auto optimizer
stats collection
Job
ORA$AT_OS_OPT_SY_<n>
Program
GATHER_STATS_PROG
Action
dbms_stats.
gather_database_stats_job_proc
MO-FR jeweils 22 Uhr – 4 Stunden
SA jeweils 6 Uhr – 20 Stunden
API über
DBMS_AUTO_TASK_ADMIN
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 8 von XX

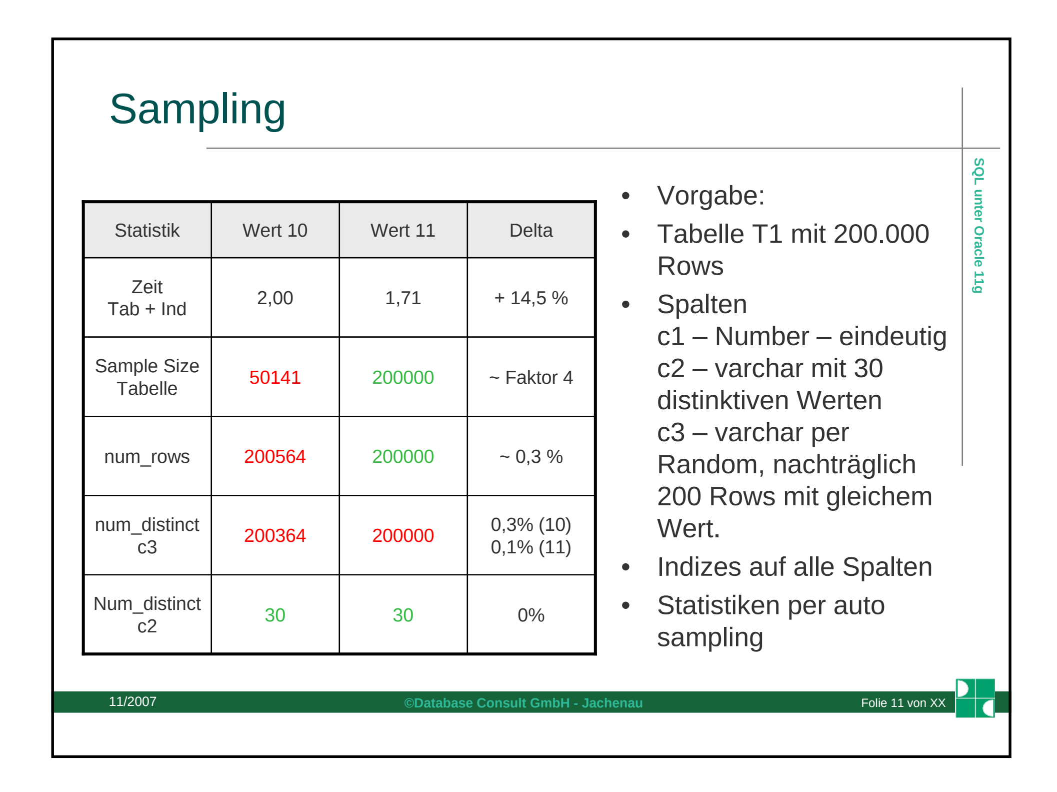
Automatische Statistik Sammlung 11g

```
BEGIN
 dbms_auto_task_admin.disable(
  client_name => 'auto optimizer stats collection',
  operation => NULL, window_name => NULL);
END;
BEGIN
 dbms_auto_task_admin.disable(
  client_name => 'auto optimizer stats collection',
  operation => NULL, window_name => 'SUNDAY_WINDOW');
END;
BEGIN
 dbms_auto_task_immediate.gather_optimizer_stats;
END;
```

SQL unter Oracle 11g

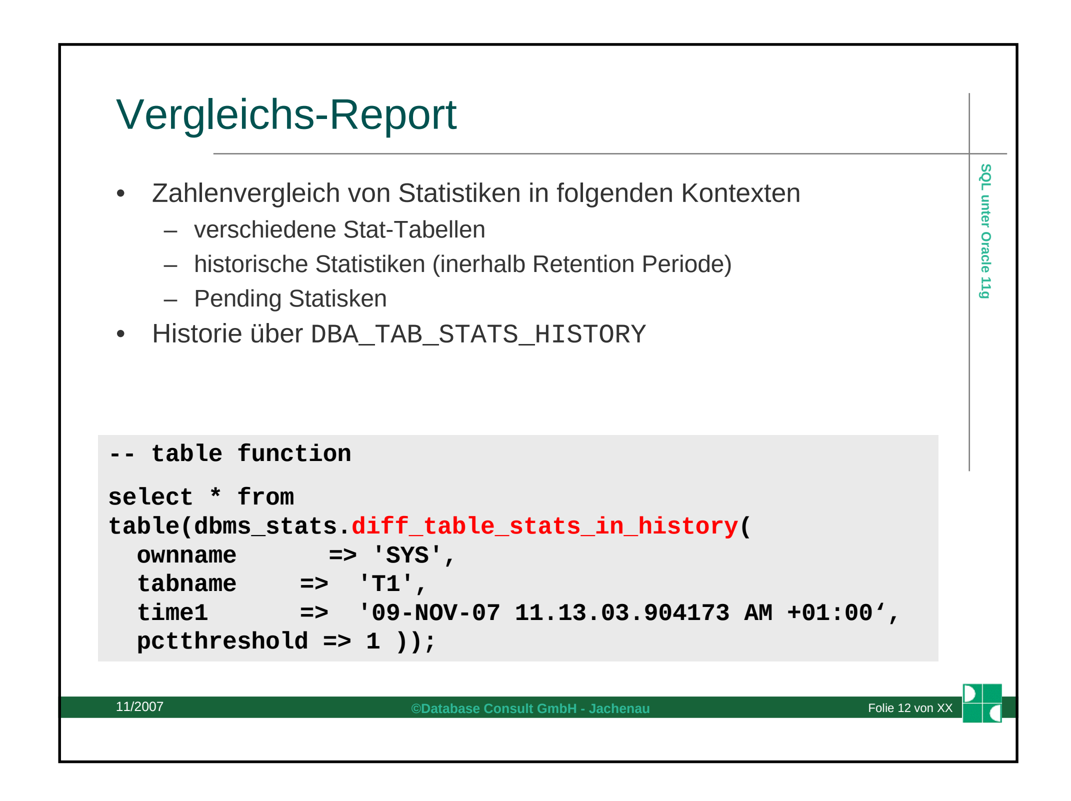
11/2007 ©Database Consult GmbH - Jachenau Folie 9 von XX




Sampling

Statistik	Wert 10	Wert 11	Delta
Zeit Tab + Ind	2,00	1,71	+ 14,5 %
Sample Size Tabelle	50141	200000	~ Faktor 4
num_rows	200564	200000	~ 0,3 %
num_distinct c3	200364	200000	0,3% (10) 0,1% (11)
Num_distinct c2	30	30	0%

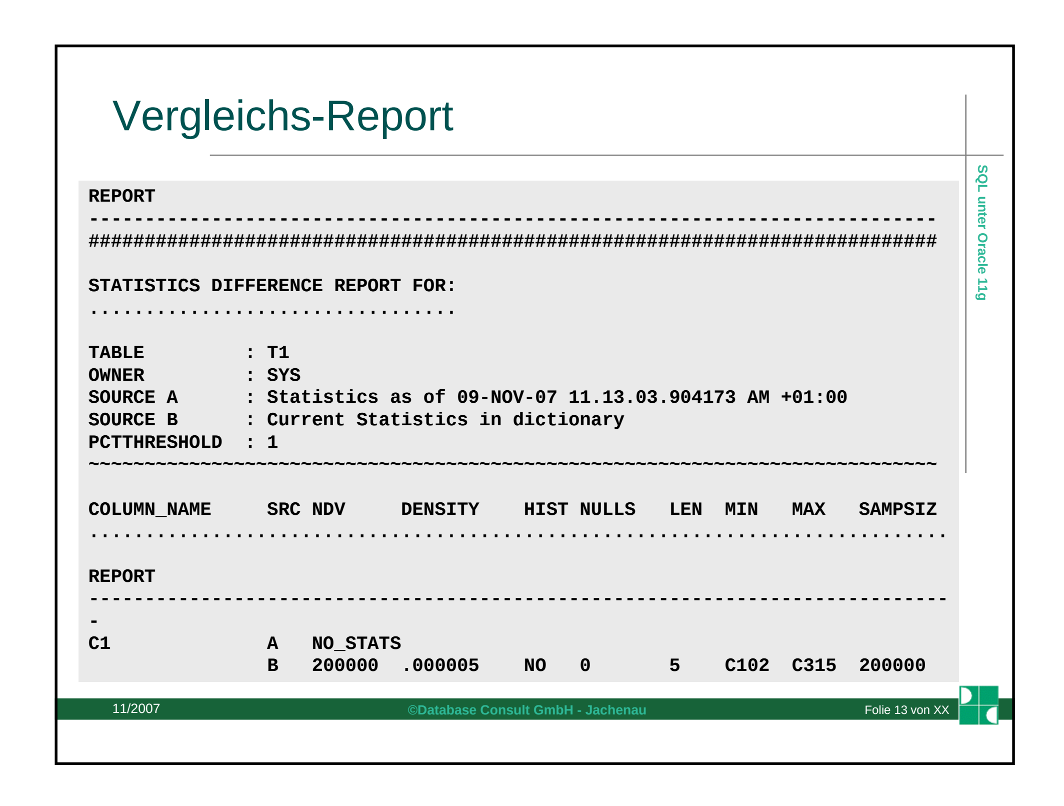
- Vorgabe:
- Tabelle T1 mit 200.000 Rows
- Spalten
c1 – Number – eindeutig
c2 – varchar mit 30 distinktiven Werten
c3 – varchar per Random, nachträglich 200 Rows mit gleichem Wert.
- Indizes auf alle Spalten
- Statistiken per auto sampling
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 11 von XX


Vergleichs-Report

- Zahlenvergleich von Statistiken in folgenden Kontexten
  - verschiedene Stat-Tabellen
  - historische Statistiken (inerhalb Retention Periode)
  - Pending Statisken
- Historie über DBA_TAB_STATS_HISTORY

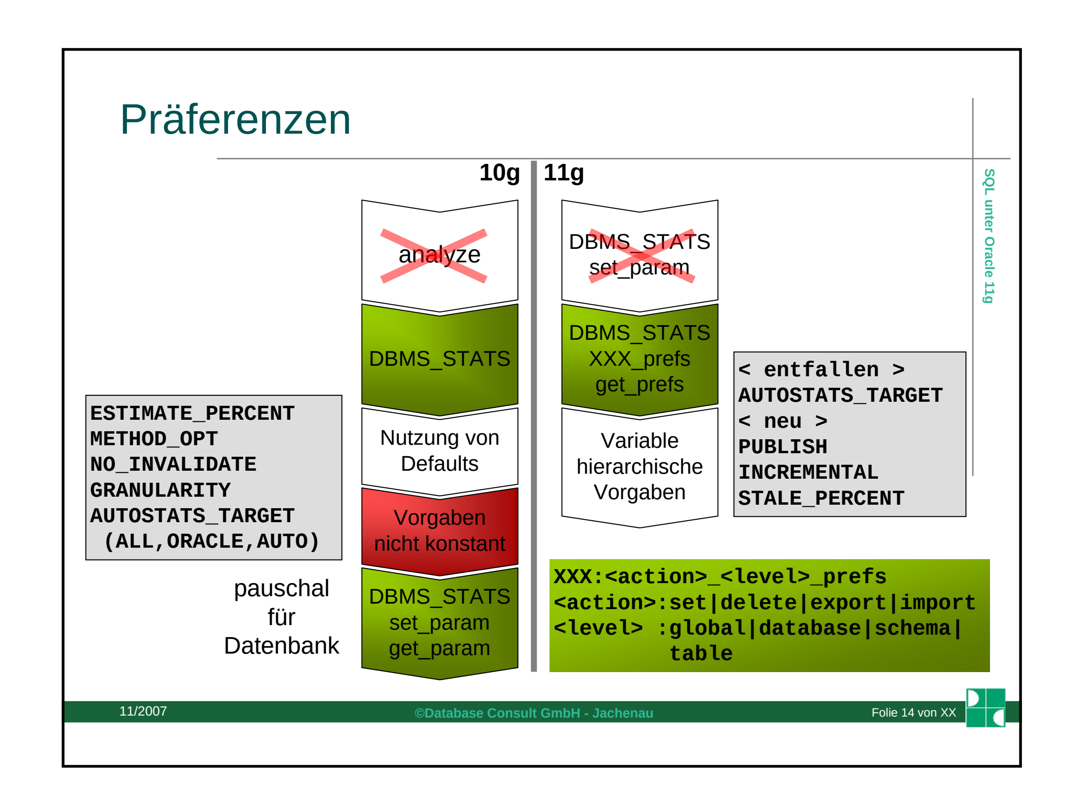
```
-- table function

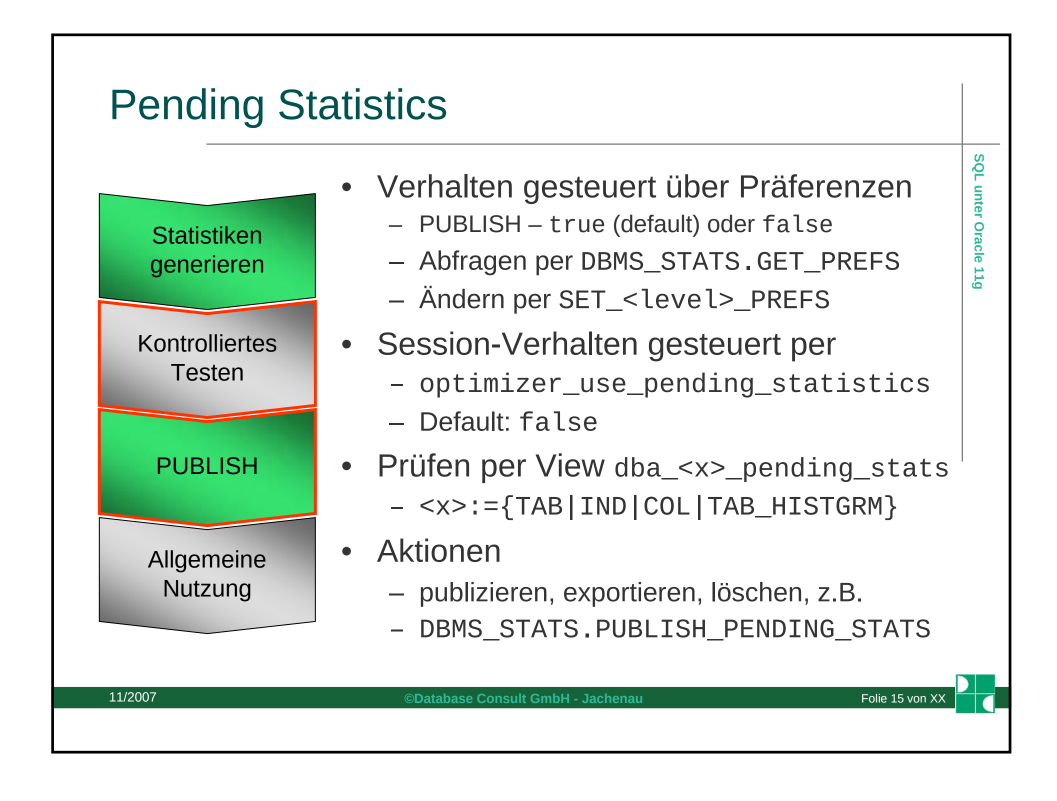
select * from
table(dbms_stats.diff_table_stats_in_history(
  ownname       => 'SYS',
  tabname     =>  'T1',
  time1       =>  '09-NOV-07 11.13.03.904173 AM +01:00‘,
  pctthreshold => 1 ));
```
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 12 von XX


Vergleichs-Report
SQL unter Oracle 11g
REPORT

##
STATISTICS DIFFERENCE REPORT FOR:
....
TABLE : T1
OWNER : SYS
SOURCE A : Statistics as of 09-NOV-07 11.13.03.904173 AM +01:00
SOURCE B : Current Statistics in dictionary
PCTTHRESHOLD : 1
~~~~~~
COLUMN_NAME SRC NDV DENSITY HIST NULLS LEN MIN MAX SAMPSIZ
....
REPORT

-
C1 A NO_STATS
B 200000 .000005 NO 0 5 C102 C315 200000
11/2007
©Database Consult GmbH - Jachenau
Folie 13 von XX


Präferenzen
SQL unter Oracle 11g
10g
11g
analyze
DBMS_STATS
Nutzung von Defaults
Vorgaben nicht konstant
DBMS_STATS set_param get_param
DBMS_STATS set_param
DBMS_STATS XXX_prefs get_prefs
Variable hierarchische Vorgaben
ESTIMATE_PERCENT
METHOD_OPT
NO_INVALIDATE
GRANULARITY
AUTOSTATS_TARGET
(ALL,ORACLE,AUTO)
pauschal für Datenbank
< entfallen >
AUTOSTATS_TARGET
< neu >
PUBLISH
INCREMENTAL
STALE_PERCENT
XXX:<action>_<level>_prefs
<action>:set|delete|export|import
<level> :global|database|schema|
table
11/2007
©Database Consult GmbH - Jachenau
Folie 14 von XX
~~~~~~



Inkrementelle Statistiken

SQL unter Oracle 11g

- Interessant für partitionierte Tabellen
  - 10g: separate Statistikgenerierung für einzelne Partitions, aber globale Statistiken immer über Gesamttabelle generiert (full scan)
  - 11g: Generierung nur für „touched partitions“
    globale Statistiken erzeugt aus Partitionsstatistiken
    kein Scan auf „non-touched partitions“
- Zahlen (Quelle Oracle):
  - Lineitem (TPC-H), 1.8 M rows, 230GB, 84 partitions, 16 cols

Touched partitions	incremental	1%	100%
1	37	797	18772
2	49	797	18772
3	65	797	18772
10	162	797	18772
20	299	797	18772

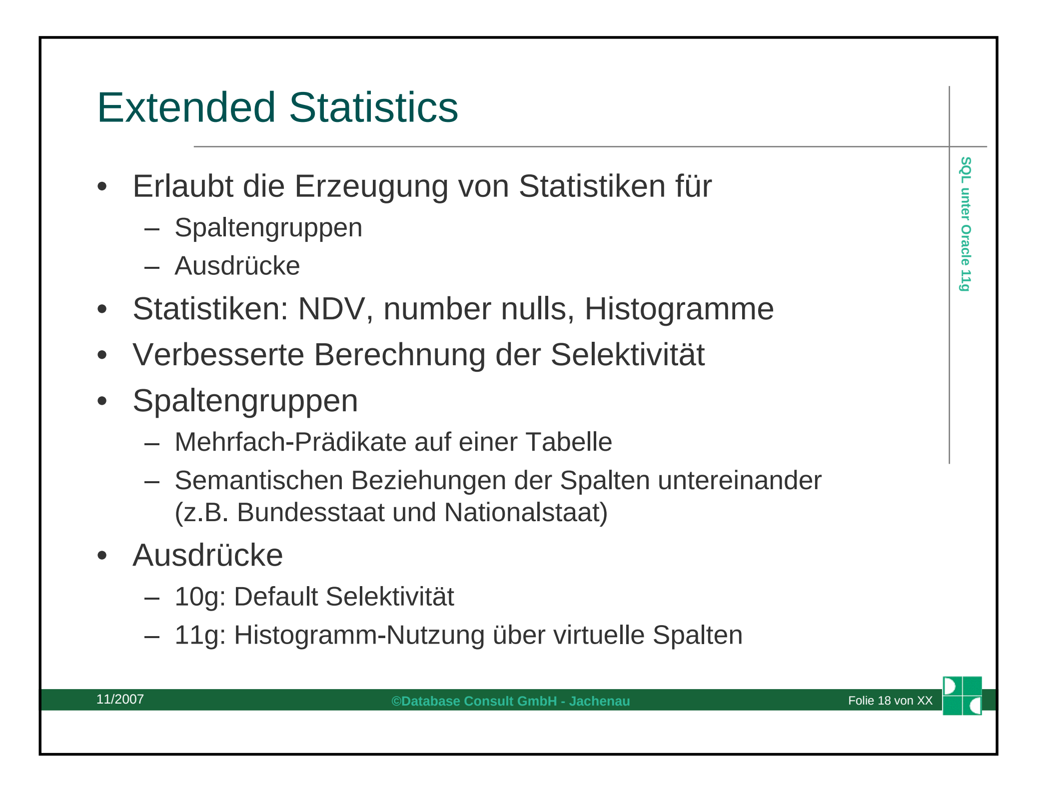
11/2007 ©Database Consult GmbH - Jachenau Folie 16 von XX

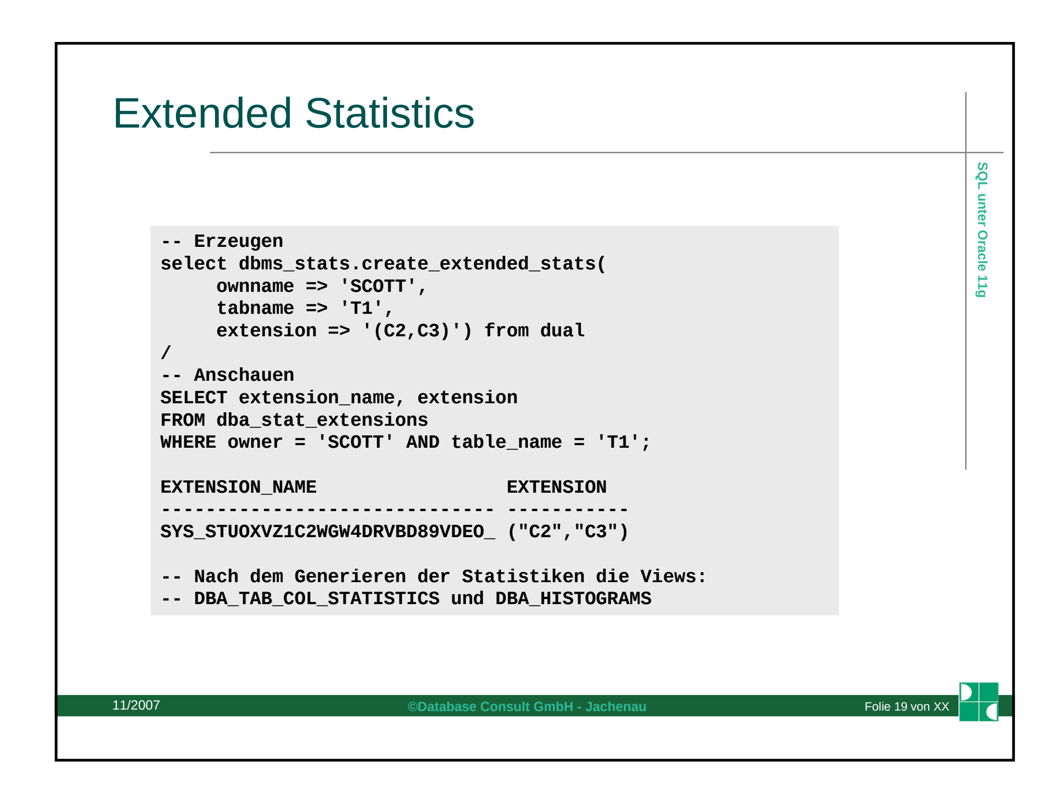
Inkrementelle Statistiken

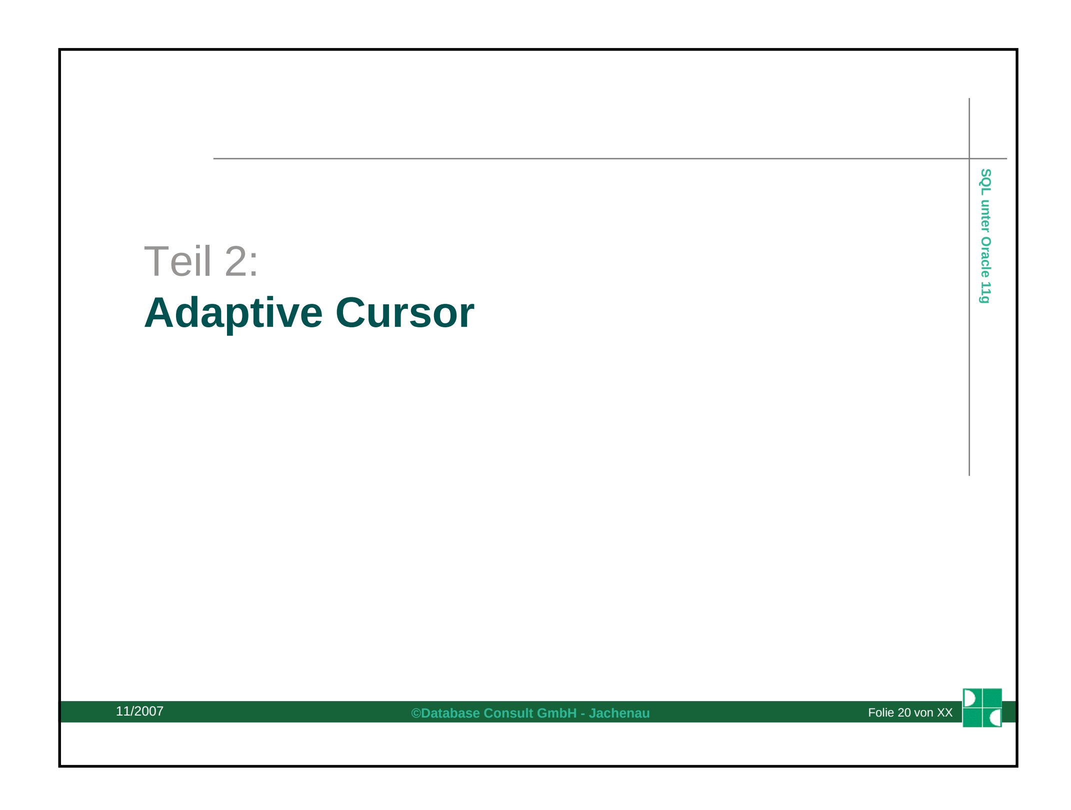
SQL unter Oracle 11g

- Voraussetzungen (AND):
- Präferenz `incremental` auf `true` (Default `false !`)
- Präferenz `publish` auf `true`
- `estimate_percent => dbms_stats.auto_sample_size`
- `granularity => 'AUTO'`

11/2007 ©Database Consult GmbH - Jachenau Folie 17 von XX




Extended Statistics
-- Erzeugen
select dbms_stats.create_extended_stats(
ownname => 'SCOTT',
tabname => 'T1',
extension => '(C2,C3)') from dual
/
-- Anschauen
SELECT extension_name, extension
FROM dba_stat_extensions
WHERE owner = 'SCOTT' AND table_name = 'T1';
EXTENSION_NAME EXTENSION
------------------------------ -----------
SYS_STUOXVZ1C2WGW4DRVBD89VDEO_ ("C2","C3")
-- Nach dem Generieren der Statistiken die Views:
-- DBA_TAB_COL_STATISTICS und DBA_HISTOGRAMS
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 19 von XX

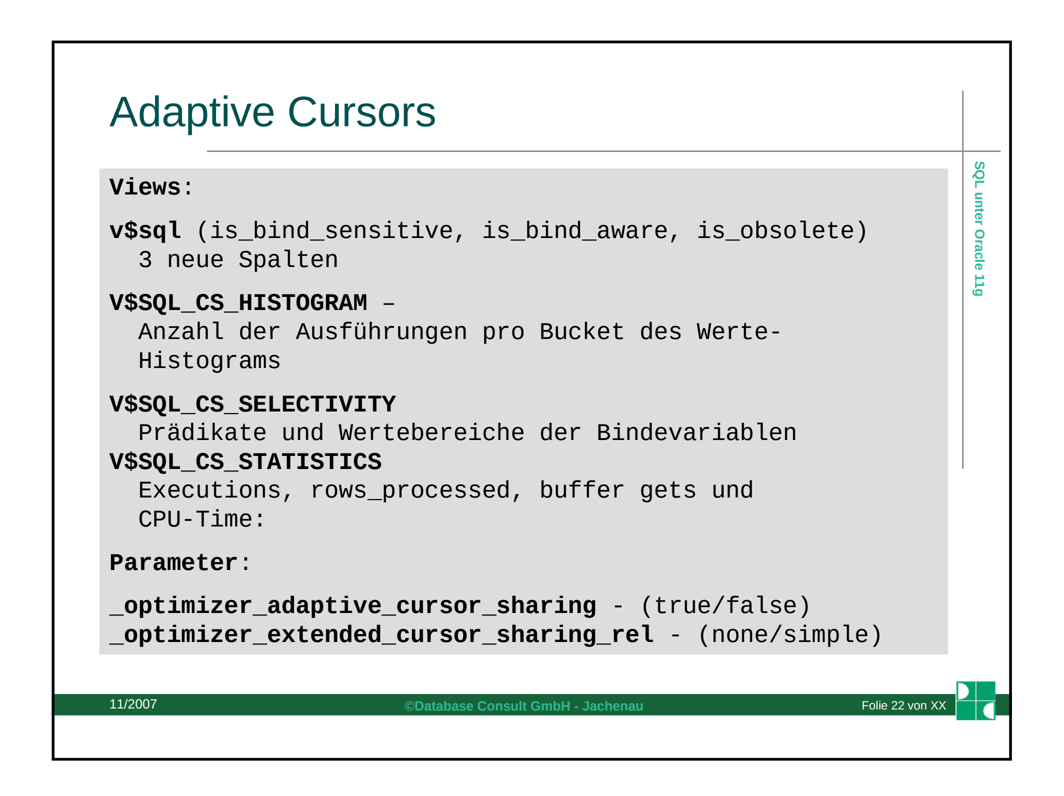

Teil 2:
Adaptive Cursor
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 20 von XX

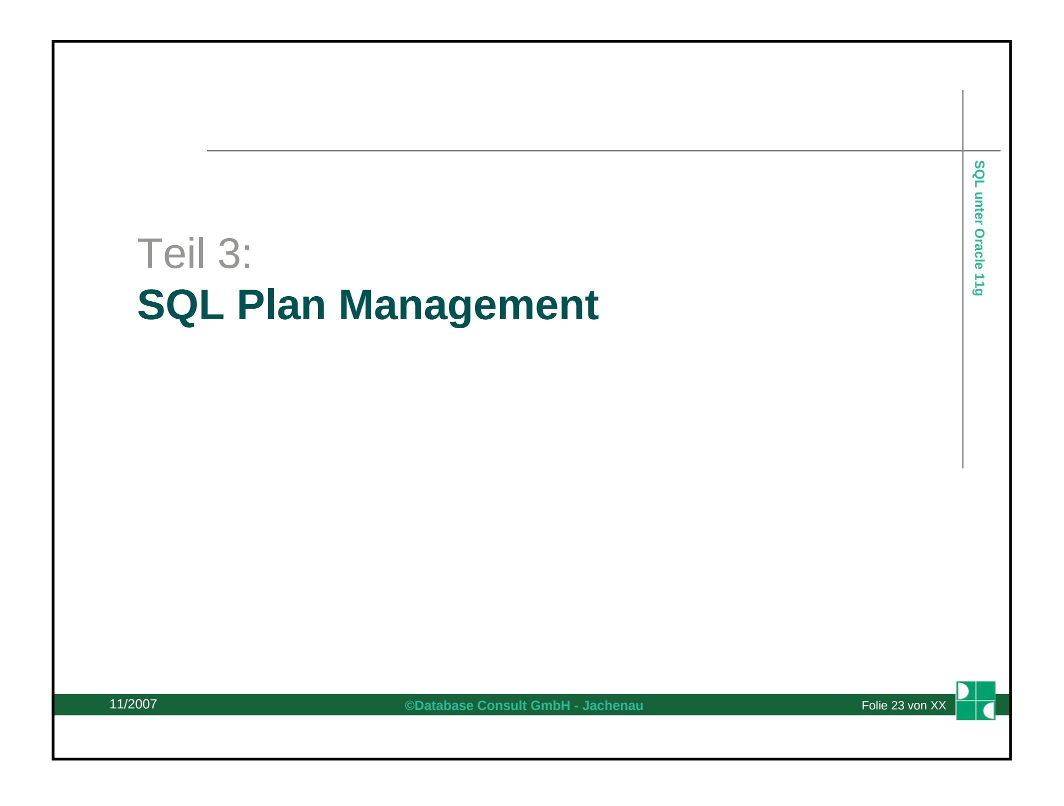
Adaptive Cursors

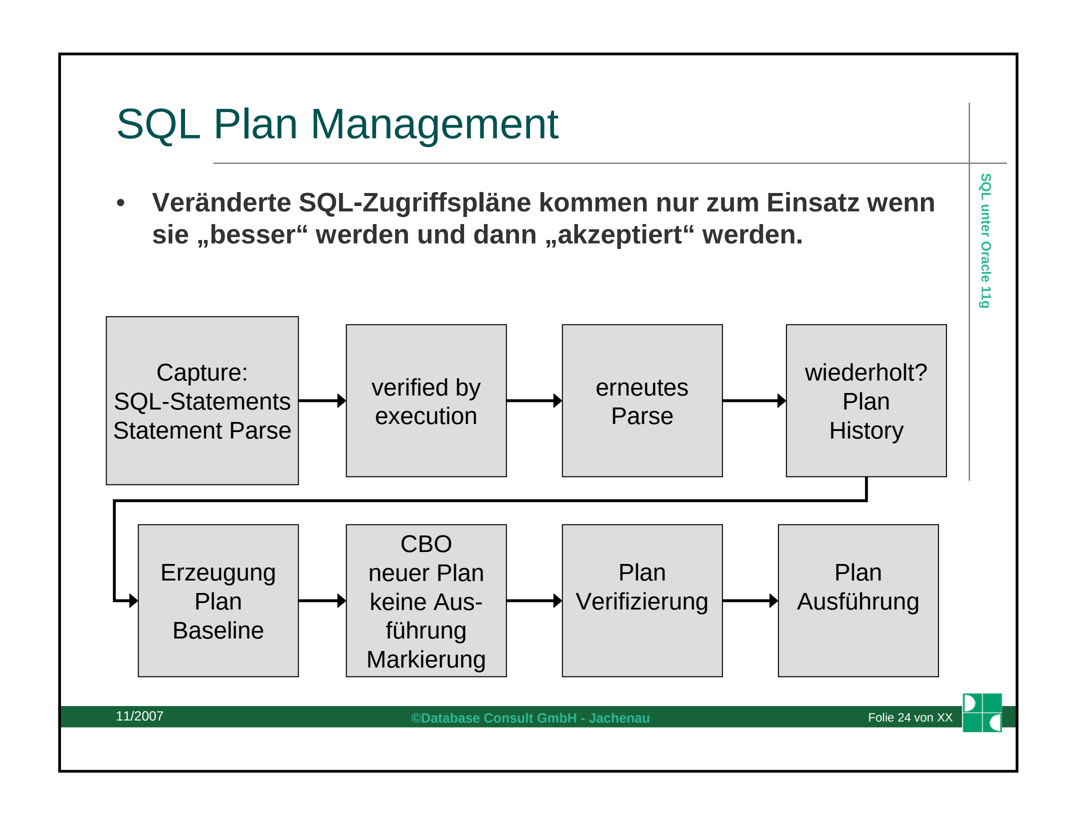
- 10g
  - bind variable peeking bei hard parse
  - bad luck bei soft parse
- 11g
  - auch: extended cursor sharing
  - Beobachtung der Werte von Bind Variablen („bind sensitive cursor“) – erstellt „bind profiles“ mit Selektivitätsbereichen
  - Bei unterschiedlicher Selektivität: Makierung - „bind aware cursor matching“ – Neues Parsing und ggf. neuer Cursor, wenn Buffer Gets besser sind, ansonsten ausführen des „alten“ Cursors.
  - ggf. mehrere Ausführungspläne pro Statement
  - eingeschaltet per Default

SQL unter Oracle 11g

11/2007 ©Database Consult GmbH - Jachenau Folie 21 von XX




SQL unter Oracle 11g
Teil 3:
SQL Plan Management
11/2007
©Database Consult GmbH - Jachenau
Folie 23 von XX


SQL Plan Management
SQL unter Oracle 11g
• Veränderte SQL-Zugriffspläne kommen nur zum Einsatz wenn sie „besser“ werden und dann „akzeptiert“ werden.
Capture: SQL-Statements Statement Parse
verified by execution
erneutes Parse
wiederholt? Plan History
Erzeugung Plan Baseline
CBO neuer Plan keine Aus-führung Markierung
Plan Verifizierung
Plan Ausführung
11/2007
©Database Consult GmbH - Jachenau
Folie 24 von XX

SQL Plan Management

- SQL Plan Baselines
  - für „wiederholt“ ausgeführte Statements – keine ad-hoc St.
  - gespeichert in SMB (SQL management Base - SYSAUX) in Form von „plan baselines“
    - SQL text, outline, bind variables, und compilation environment
  - automatisch über `optimizer_capture_sql_plan_baselines` (default FALSE) – View DBA_SQL_PLAN_BASELINES
  - oder explizit geladen (SQL Sets/AWR, Cursor Cache, Staging Table)
  - nutzbar über `optimizer_use_sql_plan_baselines`
  - Schnittstellen EM und DBMS_SPM API

SQL unter Oracle 11g

11/2007 ©Database Consult GmbH - Jachenau Folie 25 von XX



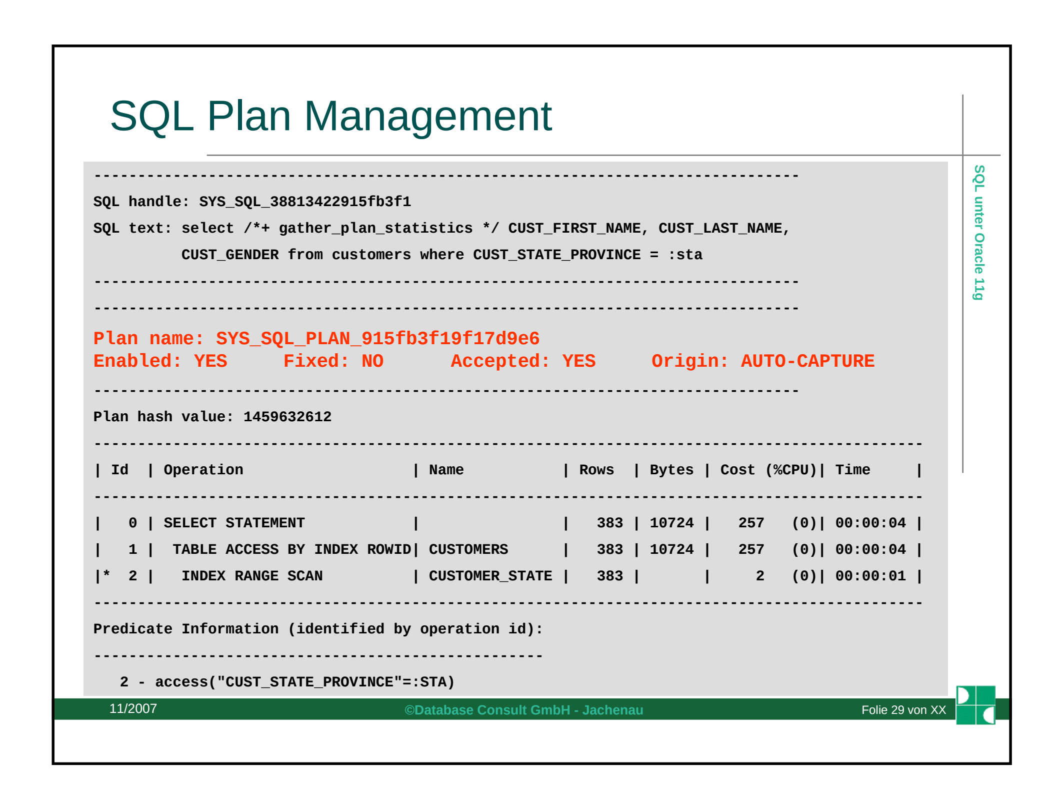
SQL Plan Management

- Neue Pläne werden als „non-accepted“ der Baseline hinzugefügt und als accepted eingestuft, wenn sie erfolgreich verifiziert werden können
- Konfiguration: %-Anteil von SYSAUX, Purge-Periode nicht genutzter Pläne (plan retention)

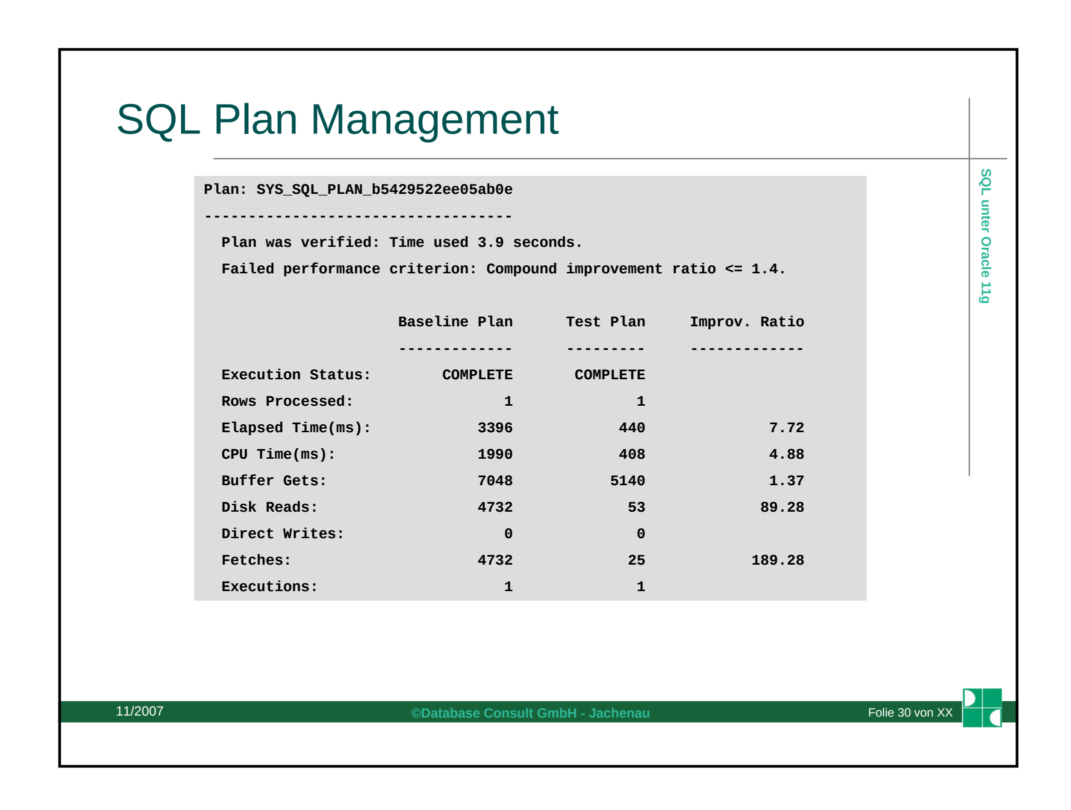
```
SET SERVEROUTPUT ON
SET LONG 10000
DECLARE
 report clob;
BEGIN
 report := DBMS_SPM.EVOLVE_SQL_PLAN_BASELINE(
           sql_handle => 'SYS_SQL_593bc74fca8e6738');
 DBMS_OUTPUT.PUT_LINE(report);
END;
```


SQL Plan Management


```
SELECT parameter_name, parameter_value
FROM DBA_SQL_MANAGEMENT_CONFIG
/
-- Konfigurieren init.ora inkl. „underscores“
BEGIN
DBMS_SPM.CONFIGURE(
’space_budget_percent’,20);
-- default 10% von SYSAUX, gültige Werte 1 – 50, ALERT.LOG
-- plan_retention_weeks –> default 53 -> Werte 5 – 523
-- gilt für nicht genutzte Pläne!
END;
/
SELECT sql_handle, plan_name, enabled, accepted, fixed
from DBA_SQL_PLAN_BASELINES;
select * from table(
dbms_xplan.display_sql_plan_baseline(
sql_handle=>'SYS_SQL_209d10fabbedc741',
format=>'basic'));
```

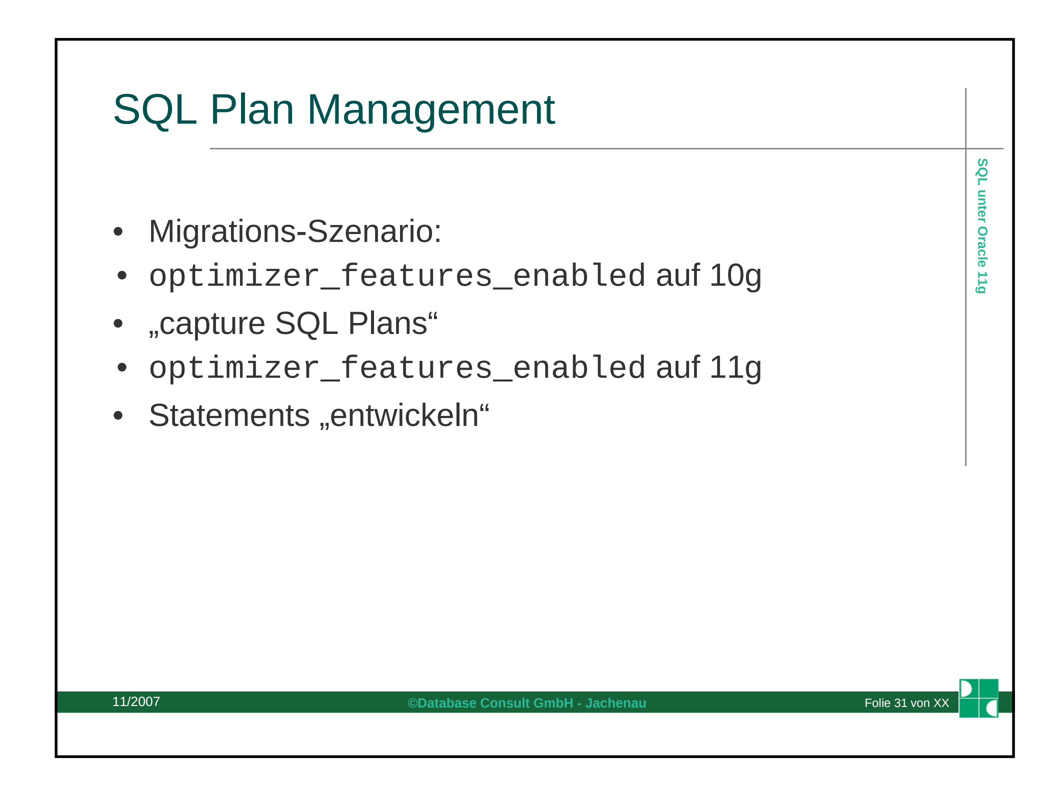


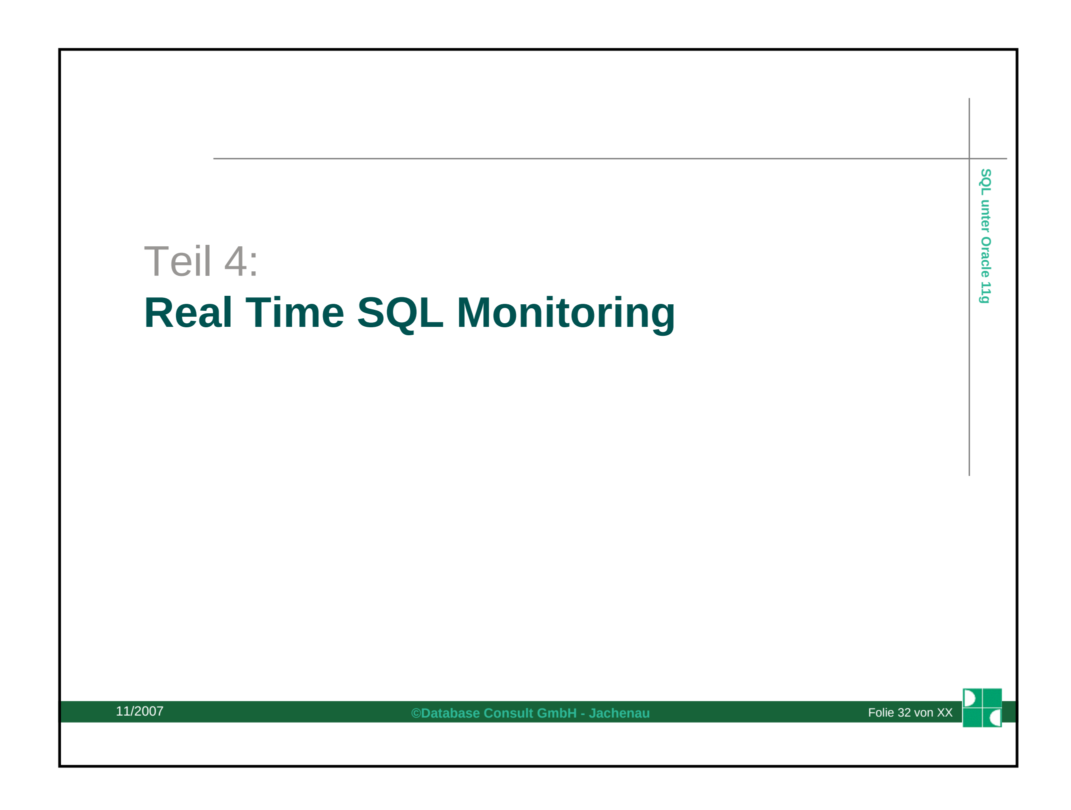

```
---------------------------------------------------------------------------
SQL handle: SYS_SQL_38813422915fb3f1
SQL text: select /*+ gather_plan_statistics */ CUST_FIRST_NAME, CUST_LAST_NAME,
          CUST_GENDER from customers where CUST_STATE_PROVINCE = :sta
---------------------------------------------------------------------------
---------------------------------------------------------------------------
Plan name: SYS_SQL_PLAN_915fb3f19f17d9e6
Enabled: YES     Fixed: NO      Accepted: YES     Origin: AUTO-CAPTURE
---------------------------------------------------------------------------
Plan hash value: 1459632612
-----------------------------------------------------------------------------------------------
| Id  | Operation                   | Name           | Rows  | Bytes | Cost (%CPU)| Time     |
-----------------------------------------------------------------------------------------------
|   0 | SELECT STATEMENT            |                |   383 | 10724 |   257   (0)| 00:00:04 |
|   1 |  TABLE ACCESS BY INDEX ROWID| CUSTOMERS      |   383 | 10724 |   257   (0)| 00:00:04 |
|*  2 |   INDEX RANGE SCAN          | CUSTOMER_STATE |   383 |       |     2   (0)| 00:00:01 |
-----------------------------------------------------------------------------------------------
Predicate Information (identified by operation id):
---------------------------------------------------
   2 - access("CUST_STATE_PROVINCE"=:STA)
```




```
Plan: SYS_SQL_PLAN_b5429522ee05ab0e
-----------------------------------
  Plan was verified: Time used 3.9 seconds.
  Failed performance criterion: Compound improvement ratio <= 1.4.

                      Baseline Plan      Test Plan     Improv. Ratio
                      -------------      ---------     -------------
  Execution Status:        COMPLETE       COMPLETE
  Rows Processed:                 1              1
  Elapsed Time(ms):            3396            440              7.72
  CPU Time(ms):                1990            408              4.88
  Buffer Gets:                 7048           5140              1.37
  Disk Reads:                  4732             53             89.28
  Direct Writes:                  0              0
  Fetches:                     4732             25            189.28
  Executions:                     1              1
```


SQL Plan Management
Migrations-Szenario:
optimizer_features_enabled auf 10g
„capture SQL Plans“
optimizer_features_enabled auf 11g
Statements „entwickeln“
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 31 von XX


Teil 4:
Real Time SQL Monitoring
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 32 von XX

SQL Real-Time Monitoring

- Beobachtung von SQL zur Laufzeit
- Statistiken für einen einzigen SQL-Lauf – nicht kumuliert
- Kriterien
  - parallel queries, parallel DML or parallel DDL
  - serielle Ausführung > 5s CPU oder IO Zeit
- Globale SQL Statistiken per **V$SQL_MONITOR**
- Plan level statistics (#rows, memory, temp space, start/end date) per **V$SQL_PLAN_MONITOR**
- Nach Laufzeitende werden Statistiken noch wenige Minuten erhalten (ca. 5)
- Bericht per DBMS_SQLTUNE.REPORT_SQL_MONITOR();

SQL Real-Time Monitoring

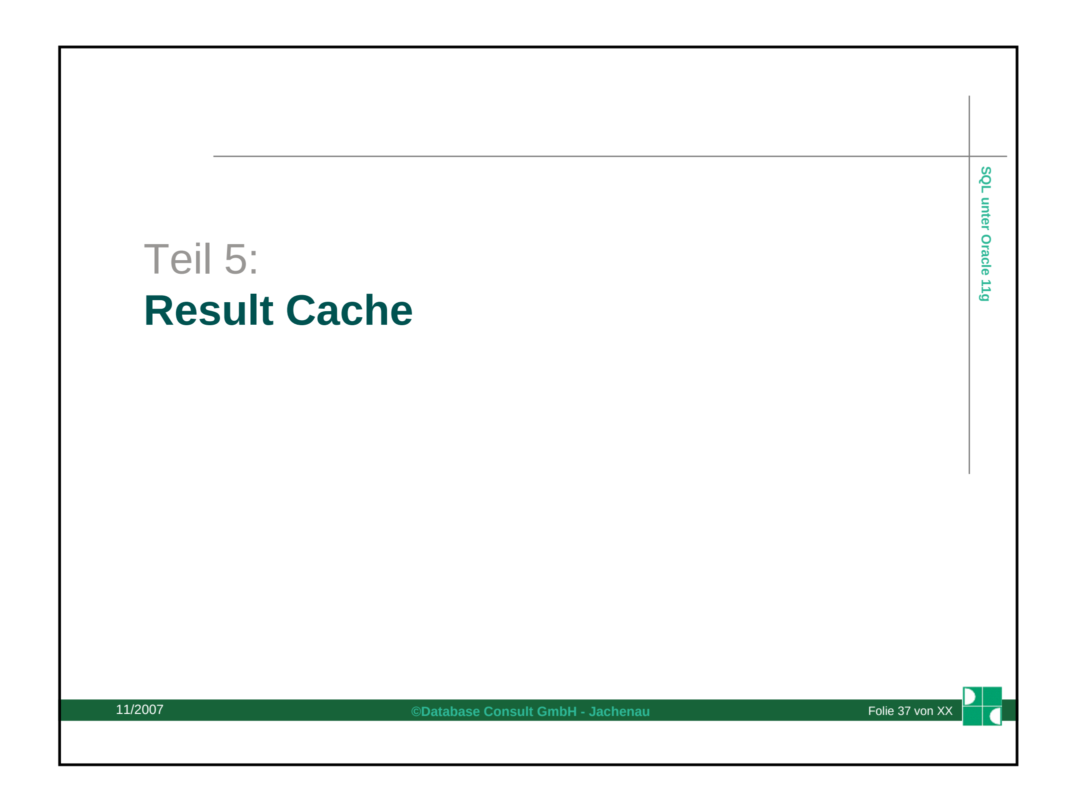

```
------------------------------------------------
Global Information
Status : EXECUTING
Instance ID : 1
Session ID : 980
SQL ID : br4m75c20p97h
SQL Execution ID : 16777219
Plan Hash Value : 2992965678
Execution Started : 06/07/2007 08:36:42
First Refresh Time : 06/07/2007 08:36:46
Last Refresh Time : 06/07/2007 08:40:02
---------------------------------------------------------------------
------
| Elapsed | Cpu | IO | Application | Other | Buffer | Reads | Writes
|
| Time(s) | Time(s) | Waits(s) | Waits(s) | Waits(s) | Gets | | |
---------------------------------------------------------------------
------
...
SQL Plan Monitoring Details
=====================================================================
```

Active Session History

SQL unter Oracle 11g

- v$active_session_history
- Bessere Identifizierung von ausgeführten SQL-Statements und wiederholten Ausführungen
  - SQL_EXEC_ID, SQL_EXEC_START
  - join mit V$SQL_MONITOR
- Identifizierung einzelner Operationen
  - SQL_PLAN_LINE_ID
  - SQL_PLAN_OPERATION
- aktuelle Operationen
  - IN_PARSE
  - IN_PLSQL_EXECUTION etc.

11/2007 ©Database Consult GmbH - Jachenau Folie 35 von XX



Result Cache

- Chaching von Result Sets, nicht von Blöcken -> Speichereffizienz
- „just-in-time materialized view“ - Teil des Shared Pool - Instanz
- Gültig bis abhängige Objekte modifiziert werden
- 3 Ebenen implementiert
  - Server Side + Client Side – OCI-Calls + PL/SQL function results

```
-- Parameter
result_cache_max_size
result_cache_max_result (Prozentsatz eines Results)
result_cache_mode = { MANUAL | FORCE }
select /*+ result_cache */ .... /*+ no_result_cahce */
-- Function Cache
CREATE FUNCTION x (p_x IN NUMBER) RETURN NUMBER
RESULT_CACHE RELIES_ON(table_t) AS ...
--
SELECT dbms_result_cache.status() FROM dual;
-- Diverses
V$RESULT_CACHE_STATISTICS - DBMS_RESULT_CACHE.MEMORY_REPORT
DBMS_RESULT_CACHE.FLUSH
```

SQL unter Oracle 11g

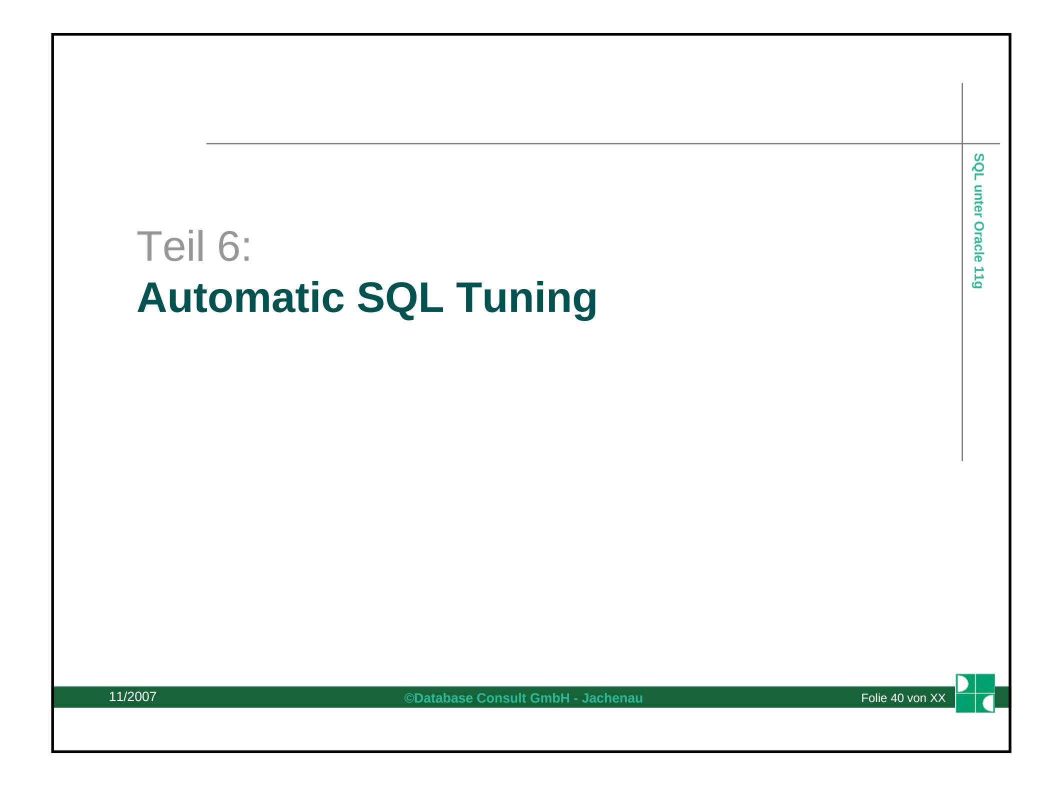
11/2007 ©Database Consult GmbH - Jachenau Folie 38 von XX

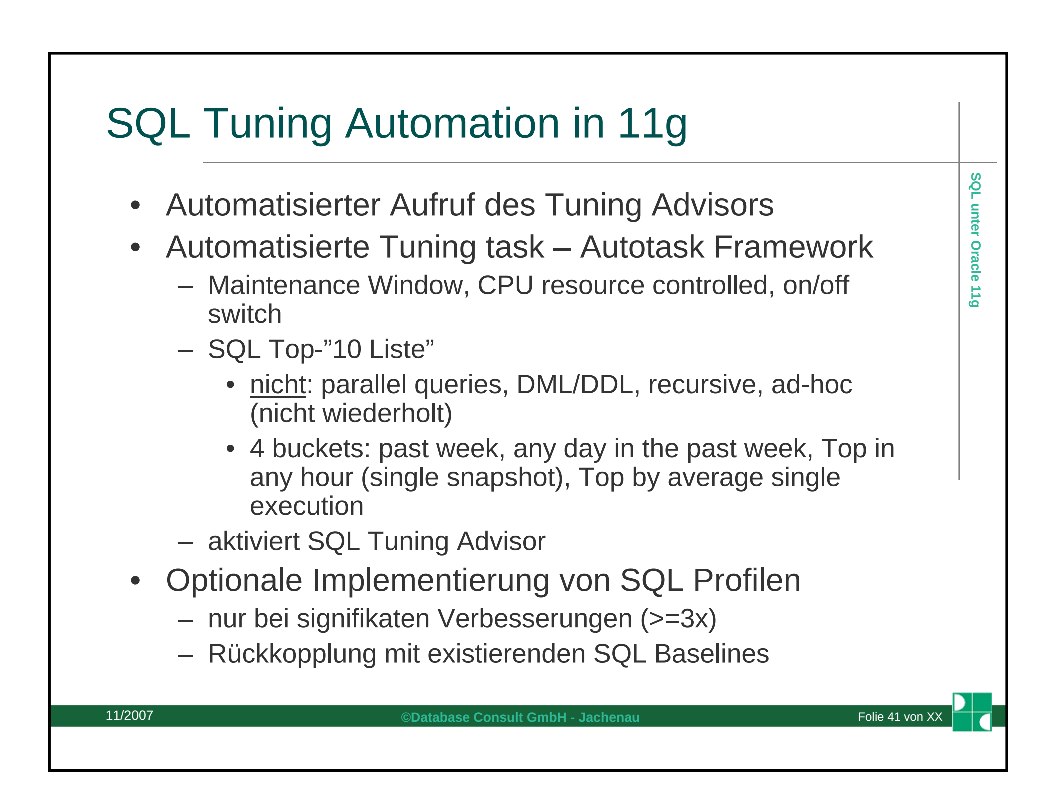
Server Result Cache

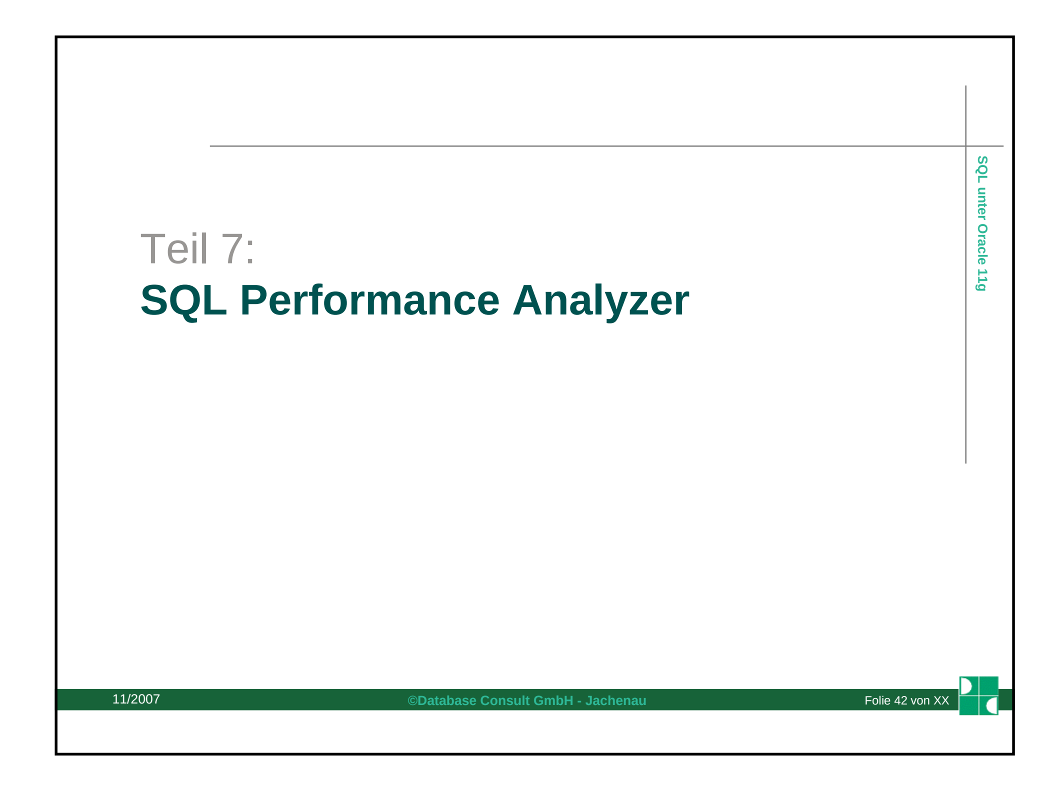
- Einschränkungen
  - Gleiche Syntax!
  - Dictionary- und temporary tables
  - CURRVAL, NEXTVAL
  - SQL functions current_date, current_timestamp, local_timestamp,
  - userenv/sys_context (mit non-constant Variablen), sys_guid, sysdate, sys_timestamp
  - Non-deterministic PL/SQL Funktionen
  - keine Subqueries
- Parametrierte Results
  - Bindevariablen als Wert!

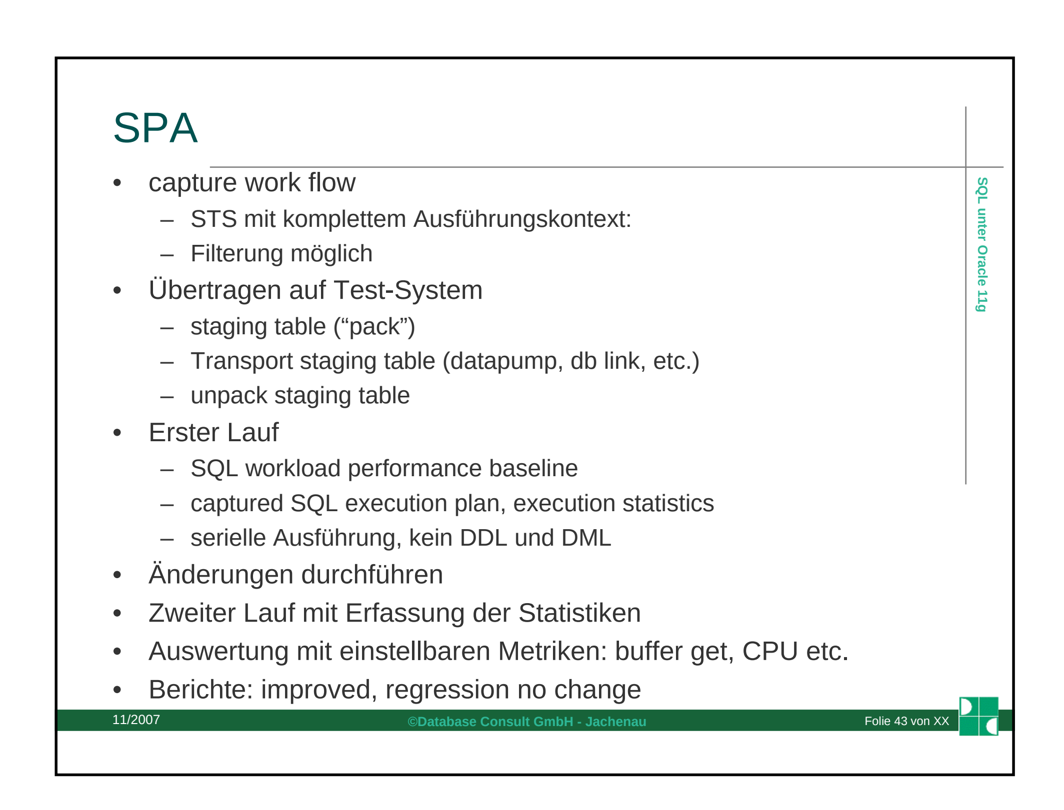
SQL unter Oracle 11g

11/2007 ©Database Consult GmbH - Jachenau Folie 39 von XX


Teil 6:
Automatic SQL Tuning
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 40 von XX


SQL Tuning Automation in 11g
• Automatisierter Aufruf des Tuning Advisors
• Automatisierte Tuning task – Autotask Framework
– Maintenance Window, CPU resource controlled, on/off switch
– SQL Top-"10 Liste"
• nicht: parallel queries, DML/DDL, recursive, ad-hoc (nicht wiederholt)
• 4 buckets: past week, any day in the past week, Top in any hour (single snapshot), Top by average single execution
– aktiviert SQL Tuning Advisor
• Optionale Implementierung von SQL Profilen
– nur bei signifikaten Verbesserungen (>=3x)
– Rückkopplung mit existierenden SQL Baselines
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 41 von XX


Teil 7:
SQL Performance Analyzer
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 42 von XX

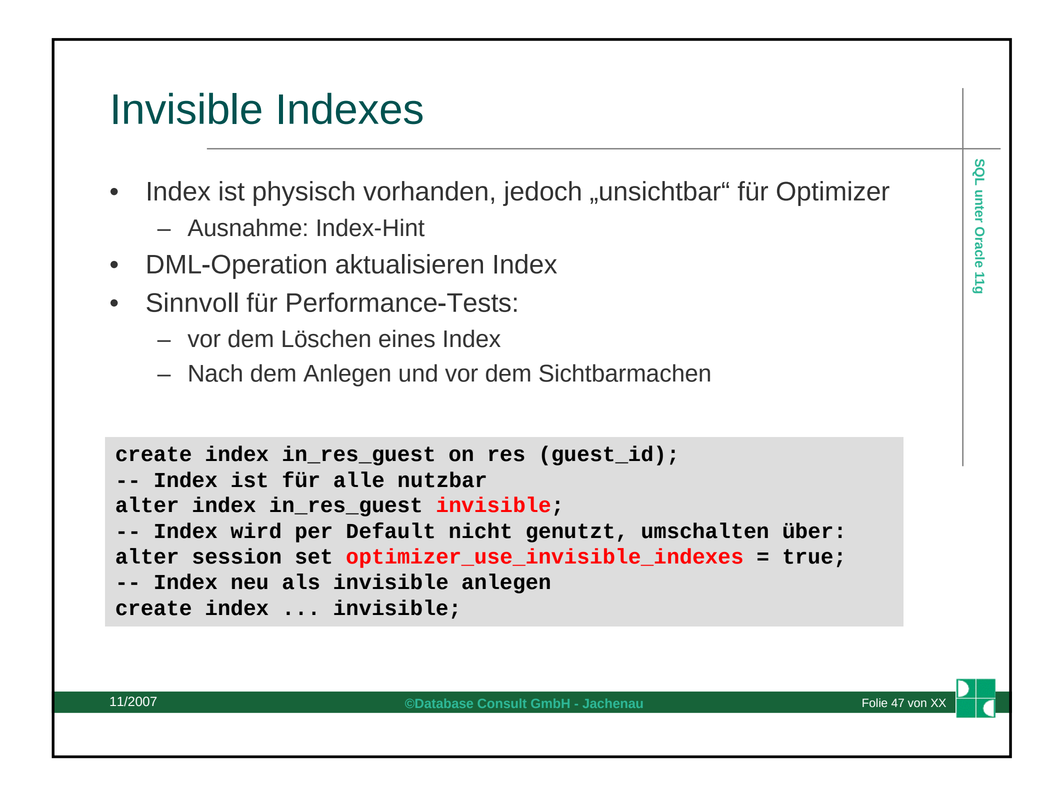

SPA
• capture work flow
– STS mit komplettem Ausführungskontext:
– Filterung möglich
• Übertragen auf Test-System
– staging table (“pack”)
– Transport staging table (datapump, db link, etc.)
– unpack staging table
• Erster Lauf
– SQL workload performance baseline
– captured SQL execution plan, execution statistics
– serielle Ausführung, kein DDL und DML
• Änderungen durchführen
• Zweiter Lauf mit Erfassung der Statistiken
• Auswertung mit einstellbaren Metriken: buffer get, CPU etc.
• Berichte: improved, regression no change
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 43 von XX

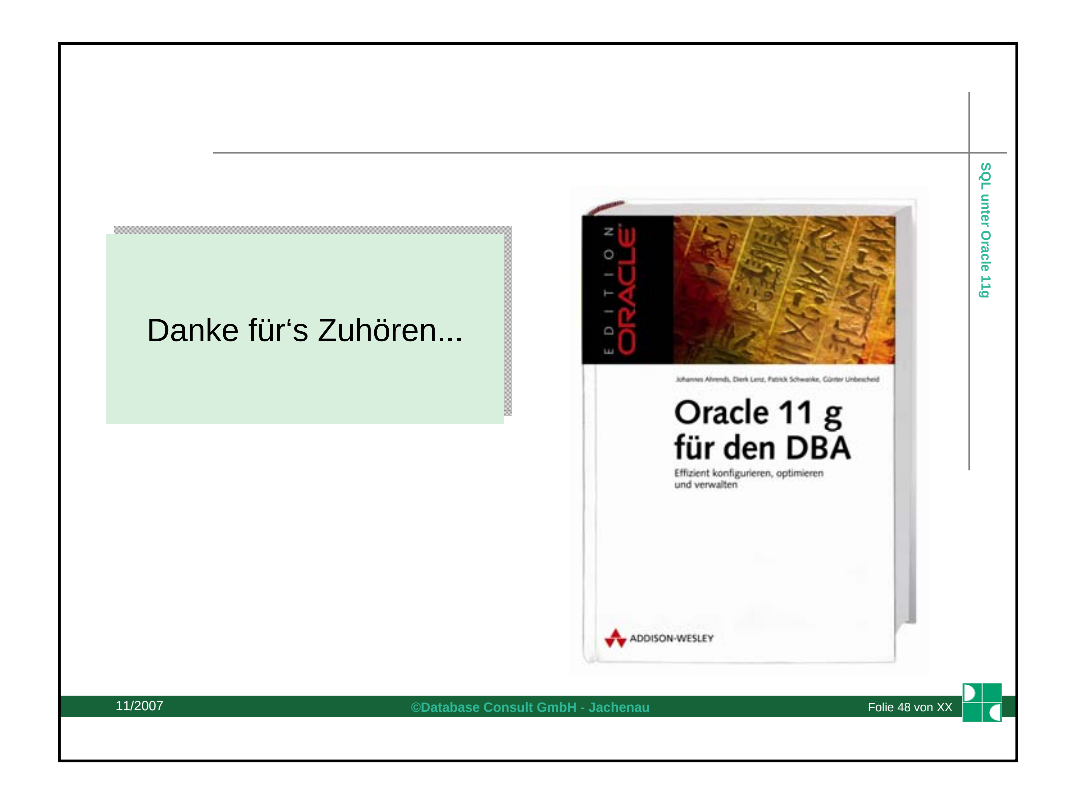
SQL Repair Advisor

- Fehler bei der Ausführung von SQL
- Incident im ADR
- Generierung von Vorschlägen
  - SQL Patch für ein Statement
  - ggf. Info-Pakete für Support vorbereiten

SQL unter Oracle 11g

11/2007 ©Database Consult GmbH - Jachenau Folie 45 von XX




Danke für's Zuhören...
Oracle 11 g
für den DBA
Effizient konfigurieren, optimieren
und verwalten
EDITION
ORACLE
ADDISON-WESLEY
SQL unter Oracle 11g
11/2007
©Database Consult GmbH - Jachenau
Folie 48 von XX